实现 sm2 实验报告

网络空间安全创新创业实践

赵翔正 202000460090

定义椭圆曲线上的加法

```
def addition(x1,y1,x2,y2,a,p):
    if x1==x2 and y1==p-y2:
        return False
    if x1!=x2:
        lamda=((y2-y1)*modinv(x2-x1, p))%p
    else:
        lamda=(((3*x1*x1+a)%p)*modinv(2*y1, p))%p
    x3=(lamda*lamda-x1-x2)%p
    y3=(lamda*(x1-x3)-y1)%p
    return x3,y3
```

定义数乘

```
def mutipoint(x,y,k,a,p):
    k=bin(k)[2:]
    qx,qy=x,y
    for i in range(1,len(k)):
        qx,qy=addition(qx, qy, qx, qy, a, p)
        if k[i]=='1':
            qx,qy=addition(qx, qy, x, y, a, p)
    return qx,qy
```

实现加密

```
def encrypt(m:str):
    plen=len(hex(p)[2:])
    m='0'*((4-(len(bin(int(m.encode().hex(),16))[2:])%4))%4)+bin(int(m.encode().
    klen=len(m)
    while True:
        k=randint(1, n)
        while k==dB:
            k=randint(1, n)
        x2,y2=mutipoint(xB, yB, k, a, p)
        x2,y2='{:0256b}'.format(x2),'{:0256b}'.format(y2)
        t=kdf(x2+y2, klen)
        if int(t,2)!=0:
            break
    x1,y1=mutipoint(gx, gy, k, a, p)
    x1,y1=(plen-len(hex(x1)[2:]))*'0'+hex(x1)[2:],(plen-len(hex(y1)[2:]))*'0'+he
    c1='04'+x1+y1
    c2=((klen//4)-len(hex(int(m,2)^int(t,2))[2:]))*'0'+hex(int(m,2)^int(t,2))[2:
    c3=sm3hash(hex(int(x2+m+y2,2))[2:])
    return c1,c2,c3
```

实现解密

```
def decrypt(c1,c2,c3,a,b,p):
    c1=c1[2:]
    x1,y1=int(c1[:len(c1)//2],16),int(c1[len(c1)//2:],16)
    if pow(y1,2,p)!=(pow(x1,3,p)+a*x1+b)%p:
        return False
    x2,y2=mutipoint(x1, y1, dB, a, p)
    x2,y2='{:0256b}'.format(x2),'{:0256b}'.format(y2)
    klen=len(c2)*4
    t=kdf(x2+y2, klen)
    if int(t,2)==0:
        return False
    m='0'*(klen-len(bin(int(c2,16)^int(t,2))[2:]))+bin(int(c2,16)^int(t,2))[2:]
    u=sm3hash(hex(int(x2+m+y2,2))[2:])
    if u!=c3:
        return False
    return hex(int(m,2))[2:]
```

说明：

加密要使用本地 txt 文件读取明文，其中明文如下所示

```
#待加密的消息M：encryption standard
#消息M的16进制表示：656E63 72797074 696F6E20 7374616E 64617264
```

加密后的结果如下所示：

```
'''
输出密文C= C1//C2//C3：
04245C26 FB68B1DD DDB12C4B 6BF9F2B6 D5FE60A3 83B0D18D 1C4144AB F17F6252
E776CB92 64C2A7E8 8E52B199 03FDC473 78F605E3 6811F5C0 7423A24B 84400F01
B8650053 A89B41C4 18B0C3AA D00D886C 00286467 9C3D7360 C30156FA B7C80A02
76712DA9 D8094A63 4B766D3A 285E0748 0653426D
'''
```